

Course Overview

ACTL3143 & ACTL5111 Deep Learning for Actuaries
Patrick Laub



Your lecturer

My name is Patrick Laub. My background:

- Bachelor in Software engineering / Mathematics (UQ)
- PhD in Applied Probability (Denmark & UQ)
- Post-doc in Lyon, France
- Post-doc in Melbourne, Australia



Course objectives

Artificial intelligence and *deep learning* for actuaries.

You will be able to:

- explain common neural network architectures,
- create deep learning models (in Keras) to solve actuarial data science problems,
- gain experience with practical computational tools (e.g. Python).



Moodle & Ed Forum

Three Moodle pages (3143, 5111, & combined).

These have:

- lecture slides & recordings,
- assessment details,
- Ed forum.

Please ask questions on the Ed forum.

If it is something confidential, then email me.



Lecture plans

1. Artificial Intelligence & Python
2. Deep Learning with Keras
3. **(Pub. Hol.)** Tabular Data
4. Computer Vision
5. Natural Language Processing
6. *Away for flexibility week*
7. Interpretability & Uncertainty
8. Recurrent Neural Networks
9. Generative Networks
10. Advanced Topics

In Week 3, attend or watch on Tuesday.



Contact hours

The lectures are 2 hours each week.

The tutorials are a mix of 'tutorials' and 'labs'. Tutorials will cover more of the theory and are examinable. Labs will cover practical Python and computational topics which are useful for deep learning.

Office hours can be arranged if needed.



Tutorial plans

1. **Lab:** Introduction to Python
2. **Tut:** Forward pass and batch optimisation
3. **Lab:** Git and GitHub
4. **Tut:** Backpropagation
5. **Lab:** Data visualisation
6. *Away for flexibility week*
7. **Tut:** Interpretability & Uncertainty
8. **Lab:** Project help
9. **Lab:** Project help
10. **Tut:** Exam preparation



Assessment

1. StoryWall (30%)
2. Project (40%)
3. Exam (30%)



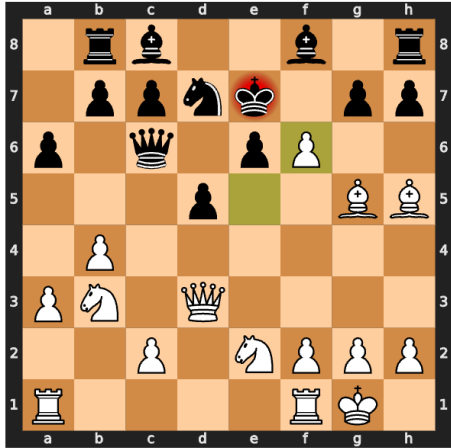
StoryWall

- StoryWall #0 is to introduce yourself, **due on Friday in Week 1** worth 2.5%.
- StoryWall #1 to #7 are 5% each, with the best 5 of 7 being counted.
- StoryWall #8 is a reflection, due on Friday in Week 10 worth 2.5%.

Each is pass/fail. Questions released on a Monday and *normally* due the one or two weeks later on a Monday.



Example StoryWall



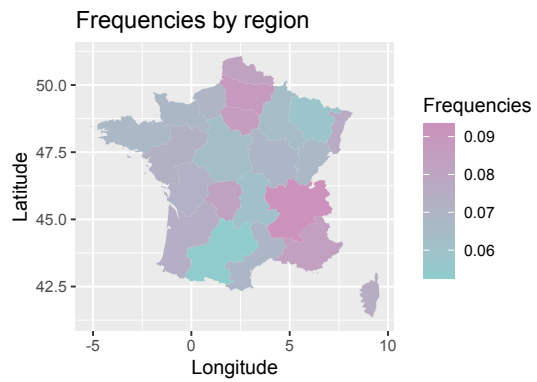
Chess AI



Undamaged



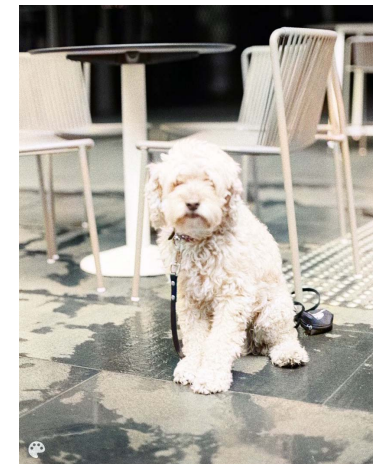
Original



French Motor Claims



Damaged



Colourised



A complete deep learning project

Individual project over the term. You will:

- specify a *supervised learning* problem,
- collect and clean the data,
- perform an exploratory data analysis (EDA),
- create a simple (non-deep learning) benchmark model,
- fit two different deep learning architectures,
- perform hyperparameter tuning,
- write a discussion of the results.



Project components

The deliverables for the project will include:

1. draft due at noon on Friday in Week 5 (10%),
2. recorded presentation due at noon on Friday in Week 8 (15%),
3. final report due at noon on Monday of Week 10 (15%).



Exam

The exam will test the concepts presented in the lectures. For example, you will be expected to:

- provide definitions for various deep learning terminology,
- suggest neural network designs to solve risk and actuarial problems,
- give advice to mock deep learning engineers whose projects have hit common roadblocks,
- find/explain common bugs in deep learning Python code.



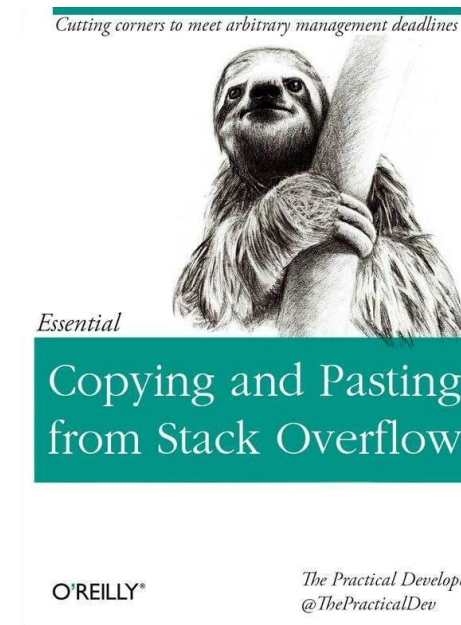
Copying code...

If you copy, tag it:

```
1 # Suppress endless warnings from Keras.
2 # Source: https://stackoverflow.com/a/38645250
3 import tensorflow as tf
4 tf.get_logger().setLevel('INFO')
```

Even if you then edit it a little:

```
1 # Create a basic Convolutional Network.
2 # Adapted from: https://www.tensorflow.org/tutorials/images/
3 model = models.Sequential()
4 model.add(layers.Input((32, 32, 3)))
5 model.add(layers.Conv2D(32, (3, 3), activation='relu'))
6 model.add(layers.MaxPooling2D((2, 2)))
7 model.add(layers.Flatten())
8 model.add(layers.Dense(64, activation='relu'))
9 model.add(layers.Dense(10))
```



Recommended reading.



Source: Anonymous (2016), [Essential Copying and Pasting from Stack Overflow](#).